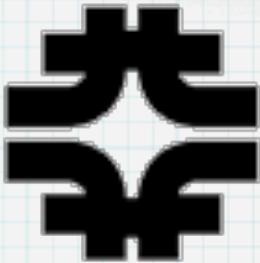




Geant 4



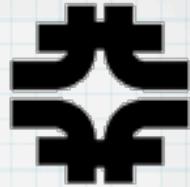
Beam Simulations the unification dk2nu

Robert Hatcher
Fermilab Computing Division

MINOS Collaboration Mtg 2013-05-04



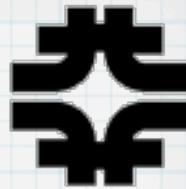
Geant 4



Unified Beam Ntuple format



- Added to nusoft repository, as “dk2nu”
 - svn checkout svn+ssh://p-nusoftart@cdcv.sfnal.gov/cvs/projects/nusoftsvn/trunk/nutools/dk2nu
 - subdir for tree structures, GENIE interface, scripts, documents
- Original Proposal: [MINOS-DocDB-9070](#)
 - initial version publicly visible on 2012-05-02
 - accompanying talk [MINOS-DocDB-9084](#)
 - also Joint NuMI Beam talk: [MINOS-DocDB-9453](#)
- Working on a document beyond “proposal”



Structure

namespace::class	tree	branch	file
bsim::Dk2Nu	dk2nuTree	dk2nu	dk2nu.h/.cxx
bsim::DkMeta	dkmetaTree	dkmeta	dkmeta.h/.cxx

● Hierarchical structure

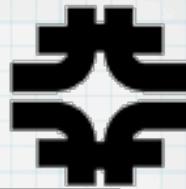
- `dk2nuTree->Draw("decay.ntype")`
- provides grouping (in browser and in class structure)
 - easier to exclude branches from reading (speedup)

● Leaves

- exact intent of the variable need description
 - **a few still unclear (e.g. pprodpx)**



Geant 4



Class Structure



Hierarchical

```
class bsim::Decay {
    int ntype;
    double nimpwt;
    ...
};

class bsim::Ancestor {
    int pdg;
    ...
};

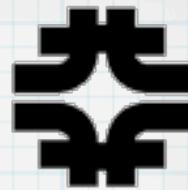
class bsim::Dk2Nu {
    bsim::Decay decay;
    vector<bsim::Ancestor> ancestor;
    ...
    int idxnu() const;
};
```

bsim::NuRay	fixed decays through specific locations {px,py,pz,E,wgt}
bsim::Decay	core info about neutrino and the decay that gave rise to it {ndecay, ntype,...} rewgt vars
bsim::Ancestor	info about the chain of particles from initial proton to the final neutrino
bsim::TgtExit	info about particles that exit the target
bsim::Traj	track points stored at special locations for plotting trajectories (not uses normally?)
bsim::Dk2Nu	all the above + job #, proton #, extensions + methods for ancestor list
bsim::Location	location where energy & weight are to be evaluated {x,y,z,name}
bsim::DkMeta	job #; POTs; general information (strings); vector of locations; extension info

- `dk2nuTree->Draw("decay.ntype:ancestor[0].pdg")`
- `dk2nuTree->Scan("ntype:ancestor[idxnu()].pdg");`
- using Dk2Nu class method `idxnu()` from library
- broken for g4minera files that overflow stored 10 ancestor limit
- `"ntype : ancestor[idxnu()].pdg : overflow()"`



Geant 4



Access to the code

- UPS version:

- ```
source /nusoft/app/alt/setup.sh
```
- ```
setup genie v3665 -q e2:debug # get GENIE + ROOT + gcc setup
```
- ```
setup dk2nu test5 -q e2:debug # new version 2013-04-10
```

- Code in “nusoft” SVN repository (visible in code browser):

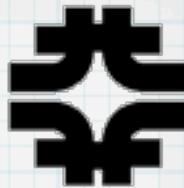
- <https://cdcv.s.fnal.gov/redmine/projects/nusoftart/repository/show/trunk/nutools/dk2nu>

- Self-build:

- ```
svn checkout svn+ssh://p-nusoftart@cdcv.s.fnal.gov/cvs/projects/nusoftsvn/trunk/nutools/dk2nu
```
- ```
export DK2NU=/path/to/dk2nu
```
- ```
cd $DK2NU ; gmake # assuming you have ROOT + gcc ( + GENIE ) setup
```



Geant 4



Trying the code - interactive

- Run code to load library, read locations file

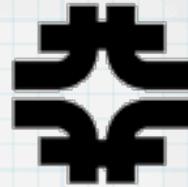
- cat \$DK2NU/etc/locations.txt
- root \$DK2NU/snippets/load_dk2nu.C \$DK2NU/snippets/test_read_locations.C
bsim::Dk2Nu dk2nu; // create one entry out of whole cloth
dk2nu.job = 42; dk2nu.decay.ntype = 14; // modify it
cout << dk2nu << endl; // have it print itself

- Run on a file (/nusoft/data/flux/dk2nu/test2)

- root \${DK2NU}/snippets/load_dk2nu.C /nusoft/data/flux/dk2nu/test2/generic_flugg_to_dk2nu.root
~~dkmetaTree->Scan("location.name","","colszie=25");~~
doesn't show all entries due to root looping bug ([Savanna report 98899](#))
TCanvas* c1 = new TCanvas(); c1->SetLogy();
dk2nuTree->Draw("nuray[1].E","nimpwt*nuray[1].wgt"); // minos near flux
dk2nuTree->SetLineColor(kRed);
dk2nuTree->Draw("nuray[3].E","nimpwt*nuray[3].wgt","SAME"); // nova near flux



Geant 4



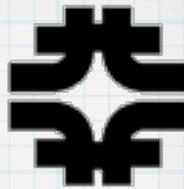
Trying the code - ancestor list



- Run on a g4minerva file (/nusoft/data/flux/dk2nu/test2)
 - root \${DK2NU}/snippets/load_dk2nu.C generic_g4minerva_to_dk2nu.root
 - dk2nuTree->Draw("@ancestor.size()"); // how many ancestors in chain?
 - "@" allows ".size()" to work on collection rather than individual elements
 - dk2nuTree->Scan("pdg[@ancestor.size()-1]", "! overflow()");
 - selects cases where there was an overflow, last entry is not neutrino (181/144078)
 - try without conditional (see most cases are neutrinos); apdg[0]=2212=proton
 - added functions: overflow(), idxnu(), idxp(), idxgp() (nu, parent, grandparent)
 - dk2nuTree->Scan("ntype:pdg[idxnu()", "! overflow()"); // nu's pdg
 - try other things



Geant 4



Trying the code - compiled

- Run code to load library, read locations file

```
• root $DK2NU/snippets/load_dk2nu.C \
 '$DK2NU/snippets/test_read_dk2nu.C+\'
 (" /nusoft/data/flux/dk2nu/test/generic_g4minerva_to_dk2nu.root " )'
```

```
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;
#include "TChain.h"
#include "dk2nu/tree/dk2nu.h"
#include "dk2nu/tree/dkmeta.h"

void test_read_dk2nu(string
pattern="generic_g4minerva*.root")
{
    TChain* cflux = new TChain("dk2nuTree");
    TChain* cmeta = new TChain("dkmetaTree");
}
```

```
cflux->AddFile(pattern.c_str());
cmeta->AddFile(pattern.c_str());
```

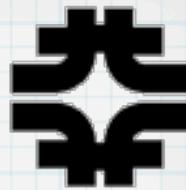
```
bsim::Dk2Nu* dk2nu = new bsim::Dk2Nu;
bsim::DkMeta* dkmeta = new bsim::DkMeta;
cflux->SetBranchAddress("dk2nu",&dk2nu);
cmeta->SetBranchAddress("dkmeta",&dkmeta);

Long64_t nflux = cflux->GetEntries();
Long64_t nmeta = cmeta->GetEntries();
cout << "nentries: " << nflux << " " << nmeta << endl;

for (Long64_t i=0;i < nflux; ++i) {
    cflux->GetEntry(i);
    if ( i < 50 ) cout << "ntype " << dk2nu->ntype << endl;
}
```



Geant 4



Converting the ntuple

- Different for each “flavor” (flugg, g4minerva, etc)

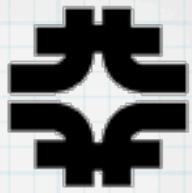
- cat \$DK2NU/etc/locations.txt
- root \$DK2NU/snippets/load_dk2nu.C \
 '\$DK2NU/convert/flugg/convert_flugg.C+("myflugg.root",42,"MINOS")'
- resulting file: myflugg_to_dk2nu.root
- Similar for g4minerva (+ eventually g4numi, g3numi)

- Conversion attempts some cross checks

- flugg compares re-calculated energy/weight to what is in the original file for Near/Far
 - thus the need to know whether it is a MINOS or NOvA generated file
- g4minerva tests whether startxyz[i] == stopxyz[i-1]
- g4minerva has odd placement for NOvA locations (?near); far is well off actuality
- g4minerva location energy+weights match up well, except “MiniBooNE” weight (energy okay)



Geant 4

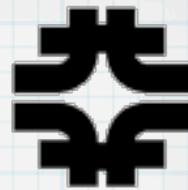


Dragons ...





Geant 4



Merging CVS Repositories



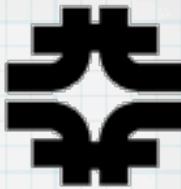
- Two phases
 - create CVS repository with all versions on branches
 - fix up CVS *,v files to get commit timestamp/authors right
- First phase: done

	RCS files	CVS files	CVS commits
MINOS	1119	2451 (65)	---
NOvA	1129	3166 (15)	1458
Minerva	601	1326 (9)	831

- changed files, added file & directories, removed files
- **not as complete as I thought on Thursday (2012-10-18)**
 - revisiting ...



Geant 4

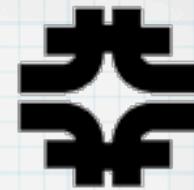


Merging CVS Repositories

- Second phase: understood
 - complex bit of bash scripting to hack up ,v files
 - “sed” is not quite the right tool, and I’m not sure what is
 - but a bit complicated ... I found something that should work
 - ```
sed --copy --in-place=.sed.bak --file=${sedscript} ${workingfile}
/^1.1.2.10$/ {
 # found revision #, append a line
 N
 # find the 2nd pattern on the next line
 /\ndate.*2012.10.16.20.03.12.*author.*rhatcher/ {
 # found it, now edit making the date, author right
 s/2012.10.16.20.03.12/2012.07.25.21.36.40/
 s/rhatcher/corwin/
 }
}
```
  - need to fix problems w/ stage 1 first



Geant 4



# Checkout code copy

- Ideally:

- ```
export CVS_RSH=ssh
```
- ```
usr/bin/cvs -d :pserver:anonymous@minoscvs.fnal.gov:/cvs/minoscvs/rep1 \
checkout -d numisoft -r nova numisoft_thedancemix
```

- But that is not right, so I took a copy of nusoft:

- ```
/usr/bin/cvs -d :pserver:anonymous@minoscvs.fnal.gov:/cvs/minoscvs/rep1 \
checkout -d numisoft numisoft_cp_nusoft
```

- -d flag, twice?

- first flag is for /usr/bin/cvs, like \$CVSROOT
- second flag is for checkout, modifies output directory to different from module name